REMARKS

Applicants respectfully traverse and request reconsideration.

As a preliminary matter, Applicants wish to thank the Examiner for the notice that Claim 16 would be allowed if rewritten in independent form.

Claims 11, 12 and 17-20 stand rejected under 35 U.S.C. § 112, 2d para., as allegedly being indefinite for failing to particularly point out an distinctly claim the subject matter which Applicants regard as the invention. As to Claim 11, the claim has been amended to correct typographical errors. It is noted that the Examiner indicated that for examination purposes it was assumed that the reference being made to first and second portions in line 8 referred to the image primitive. However, Applicants respectfully note that the first and second portions referred to in line 8 are first and second portions of the frame buffer. Claim 17 has been amended to correct several typographical errors. The amendments to the claims have been made to correct typographical errors and do not narrow or otherwise change the scope of the claims as originally filed. If the Examiner is of a different opinion Applicants respectfully request notice of the same in writing.

Claims 1-9, 11-15 and 1-20 stand rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 5,850,323 (Engstrom et al.).

Engstrom is directed to a method and system for flipping images in a window using overlay hardware of a 2D or 3D engine. The Engstrom reference is directed to a user interface that includes one or more child windows located within a parent window and occupying less than the entire display (Col. 2, ll. 1-5). In the prior art, a problem arose when this display hardware performed a screen flip wherein it flipped the entire display and not individual windows within the display. As such, Engstrom is directed to providing subscreen or child window flipping operations using overlays. Once a flip operation occurs so that the information is stored in a front buffer, the front window is then placed in the primary surface of the frame buffer so that the information may be displayed. As noted, for example, in Col. 15, the Engstrom reference describes a double buffered flipping environment. Also, Engstrom teaches to avoid modifying surface memory as the display controller is reading to display the screen. The display device interface 50 checks the state of the display hardware before attempting operations that could

cause a conflict. The display device interface appears to be a software module executing by the CPU. As noted also in Engstrom in Col. 4, the display device interface includes a software algorithm, and interfaces with the hardware abstraction layer 54 which is a hardware dependent interface. The hardware abstraction layer includes hardware-specific code. (See, e.g., ll. 46-48).

Engstrom teaches using CPU based (or system level) polling to determine whether or not it is suitable to flip between different memory locations and subsequently write a front buffer to a frame buffer memory. For example, the display device interface 50 is a software interface (driver or application) executed by the CPU to support flipping of an image in a window using overlays. The display device interface 50 acts as an interface to display hardware such as video cards (Col. 6, ll. 13-14). To support flipping in the window, a software application (hence the CPU) asks the display device interface to create a flipping structure including a front and back buffer to represent an overlay surface. In addition to the overlay structure, the software application also asks the display device interface to create a primary surface structure to represent the frame buffer. As such, the display device interface creates two flipping buffers and a primary surface structure that represent the frame buffer.

As such, the Engstrom operation appears to be similar to the prior art system described in Applicants' Background of the Invention section. For example, where a CPU polls to determine whether or not a portion of data in a frame buffer has been displayed, it can require it to be necessary to wait until the display engine indicates that all locations that the frame buffer needed to store the primitive image have been rastered. In other words, where a large triangle is to be issued for rendering, and only a small portion of the triangle is below the line currently being rastered, the polling operation can result in the display engine indicating the frame buffer is not ready. Therefore, a dispatch of the operation can be stalled even though the rendering engine could be doing useful work on most of the triangle. (See, for example, Specification, page 2, ll. 22-30).

As for Claims 1, and 3-5, the Office Action cites Engstrom at Col. 7, ll. 17-22 and Col. 20, ll. 9-17 and Col. 21, ll. 28-33. Applicants respectfully note that the display device interface mentioned in Col. 20, ll. 9-17 is a software interface executed by the CPU and as such the CPU effectively polls the display hardware before attempting operations that would cause a conflict.

In contrast, Applicants claim, among other things, enabling, by a write behind controller, storage of the image at a first memory location when the second memory location of a frame buffer indicates raster accessed data at the first memory location, and preventing, by the write behind controller, storage of the image at the first memory location when the second memory location of the frame buffer indicates the raster has not accessed data at the first memory location. Moreover, Applicants claim using frame buffer memory that can be accessed by a rasterizer. As such, among other differences, the method requires using a hardware write behind controller, which by way of example may be included as part of a 3D rendering engine or may be part of a display device controller or other suitable structure. Moreover, Applicants claim that the write behind controller determines the memory locations in a frame buffer. Accordingly, these claims are believed to be in condition for allowance.

As to Claim 2, Applicants respectfully submit that this claim adds additional novel subject matter and is also allowable as depending from an allowable base claim.

As for Claims 6 and 7, Col. 7, ll. 17-25 of Engstrom have been cited. However, this section when referring to the reading of the information and reading the image in the front buffer, appears to describe superimposing the new image in the front buffer with the primary surface at the proper location (also, the claim requires rendering an image when the image is to be stored at a first memory location of a first frame buffer). The back buffer described in this section does not appear to be the frame buffer (the primary surface structure) and as such these claims are not anticipated.

As to Claims 8 and 9, the Office Action cites Col. 4, ll. 59-62 of Engstrom. However, these lines merely indicate that a 2D or 3D graphics engine is the display hardware described in Engstrom. Applicants respectfully submit that the cited portion does not describe how graphics primitives are provided to the graphics engine when the rendering engine is storing data to a frame buffer wherein the frame buffer is being accessed by a display device controller that is providing a current image. The claim further requires that the display device controller is at a point where it has not yet accessed an address location having data associated with a current image wherein that location is between the first two address locations such that the graphics primitive is provided to the rendering engine at this point. Typically, in prior art systems, the

graphics primitive would not be submitted to a rendering engine at this point in time. Accordingly, this claim is also believed to be in condition for allowance.

Claims 11, 12 and 17-20 stand rejected based on the same rationale for Claim 1. Applicants respectfully reassert the relevant remarks made above with respect to Claim 1. Accordingly, these claims are also believed to be in condition for allowance.

Claim 10 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Engstrom in view of official notice. Applicants respectfully submit that this claim adds additional novel subject matter and is also allowable as depending upon an allowable base claim.

Attached hereto is a marked up version of the changes made to the claims by the current amendment. The attached page is captioned "Version with Markings to Show Changes Made."

Applicants respectfully submit that the claims are in condition for allowance, and an early Notice of Allowance is earnestly solicited. The Examiner is invited to telephone the below-listed attorney if the Examiner believes that a telephone conference will expedite the prosecution of the application.

Date: September 4, 2002

Respectfully submitted,

By: _____
Christopher J. Reckamp
Registration No. 34,414

VEDDER, PRICE, KAUFMAN &
KAMMHOLZ
222 N. LaSalle Street
Chicago, IL 60601
(312) 609-7599;
Facsimile: (312) 609-5005

VERSION WITH MARKINGS TO SHOW CHANGES MADE

In the claims:

Please amend Claims 1, 8, 11, 16 and 17 as follows. In particular, please substitute the below claims for the same claims with like number:

1. (Once Amended) A method for providing image data:

receiving a rendering command;

rendering an image based upon the rendering command, wherein the image is to be stored at a first memory location of a first frame buffer;

determining a second memory location representative of a raster location;

enabling, by a write behind controller, storage of the image at the first memory location when the second memory location indicates the raster has accessed data at the first memory location; and

preventing, by the write behind controller, storage of the image at the first memory location when the second memory location indicates the raster has not accessed data at the first memory location.

8. (Once Amended) A method of providing image data:

defining a graphics primitive having a first portion at X and a second portion at Y, wherein X and Y are indicative of address locations;

providing the graphics primitive to a rendering engine when the rendering engine is storing data to a frame buffer being accessed by a display device controller providing a current image, where the display device controller is yet to access an address location Z having data associated with the current [device]image and the location Z is between X and Y[.]; and

preventing tearing of the current image[;].

11. (Once Amended) A method of providing image data:

accessing a first portion of video/graphics data from a first portion of a frame buffer for display on a display device;

storing a first portion of an image primitive to the first portion of the frame buffer after the step of displaying the first portion of video/graphics data; and

prohibiting, by a write behind controller, a second portion of the image primitive from being stored to a second portion of the frame buffer after the step of storing the first portion, wherein the second portion of the frame buffer is adjacent to the first portion of the frame buffer.

16. (Once Amended) The system of claim 13, wherein the write behind raster controller includes:

a multiplexor having a first input, a second input, and an output;

a latch having an input coupled to the output of the multiplexor, and an output;

a [comparitor]comparator having a first input coupled to the output of the latch, a second input, and an output; and

an incrementor having a first input coupled to the output of the latch, and an output coupled to the first input of the multiplexor.

17. (Once Amended) A system for storing video/graphics data, the system comprising:

a rendering engine for rendering a primitive image and writing data representing the primitive image into a frame buffer;

a display device controller for reading data from the frame buffer for display; and

a write prohibit means coupled to the display device controller to receive an indication of data read by the display device controller, and coupled to the rendering [device]engine to prevent a first portion of the primitive image from being written to the frame [engine]buffer, while allowing a second portion of the primitive image to be written to the frame buffer.